

## N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM  
MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT  
CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED  
IN THE INTEREST OF MAKING AVAILABLE AS MUCH  
INFORMATION AS POSSIBLE

"JOB BUILDER" REMOTE BATCH PROCESSING SUBSYSTEM

I. G. Orlov and T. L. Orlova

(NASA-TM-76251) JOB BUILDER REMOTE BATCH  
PROCESSING SUBSYSTEM (National Aeronautics  
and Space Administration) 16 p  
HC A02/MF A01

N81-18672

CSCI 09B

Unclas  
G3/60 41535

Translation of "Podsystema distantсионnoi paketnoi obrabotki  
JOB BUILDER," USSR Academy of Sciences, Institute of Space

Research, Moscow, Report Pr-438, 1978, pp 1-15



NATIONAL AERONAUTICS AND SPACE ADMINISTRATION  
WASHINGTON, D.C. 20546  
JULY 1980

## STANDARD TITLE PAGE

1. Report No. NASA TM-76251	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle "JOB BUILDER" REMOTE BATCH PROCESSING SUBSYSTEM		5. Report Date July 1980
		6. Performing Organization Code
7. Author(s) I. G. Orlov and T. L. Orlova		8. Performing Organization Report No.
		10. Work Unit No.
9. Performing Organization Name and Address Leo Kanner Associates Redwood City, California 94063		11. Contract or Grant No. NASw-3199
		13. Type of Report and Period Covered Translation
12. Sponsoring Agency Name and Address		14. Sponsoring Agency Code
15. Supplementary Notes Translation of "Podsystema distantсионной пакетной обработки JOB BUILDER," USSR Academy of Sciences, Institute of Space Research, Moscow, Report Pr-438, 1978, pp 1-15		
16. Abstract The functions of the "JOB BUILDER" remote batch processing subsystem are described, and instructions are given for using it as a component of a display system developed by personnel of the System Programming Laboratory, Institute of Space Research, USSR Academy of Sciences.		
17. Key Words (Selected by Author(s))		18. Distribution Statement Unclassified-Unlimited
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages
		22. Price

## TABLE OF CONTENTS

Annotation

Introduction

1. Basic Operating Principles and Capabilities of the Remote Batch Processing Subsystem	1
2. Formation of Tasks	3
3. Regimes of the TASK Function	3
4. Viewing Results of Task Processing	4
5. Control Statements of the LIST Function	6
6. The TEST Function	8
Alphabetical List of Messages in the Remote Batch Processing Subsystem	10

### Annotation

The functions of the "JOB BUILDER" remote batch processing subsystem are described, and instructions are given for using it as a component of a display system developed by personnel of the System Programming Laboratory, Institute of Space Research, USSR Academy of Sciences.

### Introduction

The "JOB BUILDER" remote batch processing subsystem is designed for preparing, queuing, implementing, and viewing the results of user tasks. Tasks are input and results output by means of display terminals.

The remote batch processing subsystem is part of a display system based on BSAN displays. It runs under the control of the DOS/ES 2.0 operating system.

The user can enter his task in the input queue from the subscriber's terminal and obtain output directly at the subscriber's terminal from the output queue. However, the user cannot act on his task once it has been placed in the job queue.

## "JOB BUILDER" REMOTE BATCH PROCESSING SUBSYSTEM

I G. Orlov and T. L. Orlova

### 1. Basic Operating Principles and Capabilities of the Remote Batch Processing Subsystem

/4\*

The remote batch processing subsystem functionally consists of two groups of programs:

- 1) programs operating within the framework of the display system;
- 2) programs that supplement the DOS/ES operating system.

At the subscriber's console the user formulates his task, entering data and/or the program text both from the console and from the display system source module library. The user is accorded the possibility of editing and scrolling through the input sequence of lines. After finally formulating the text of his task, the user enters the request "PLACE TASK IN JOB QUEUE." The "BUILDER" subsystem enters in the source module library the text of the task input by the user, and places a reference to the task in the job queue. The task is assigned a number which lights up on the display screen.

Then the programs running under the display system complete their work on the given task. A task placed in the job queue will be implemented in a different section. This section is called the subordinate section, and usually has a lower priority than the display system.

---

\* Numbers in the margin indicate pagination in the foreign text.

Implementation of the task is initiated in the subordinate section by supplementary phases of the "Job Management Program."

Upon completion of any job executed in the subordinate section, the programs of the remote batch processing subsystem check whether there are jobs in the job queue. If the job queue is empty, a new job is entered from the job input device. If the job queue contains jobs ready for processing, input is accomplished from a section of a direct-access device. The entire text of the job entered from the subscriber's console is copied from the display system source module library onto a section of the direct-access device. Then all entry of the job is accomplished from the direct-access device.

System printing also is assigned to a section of the direct-access device.

Programs running under the display system of the Institute of Space Research, USSR Academy of Sciences, perform three basic functions:

/5

- 1) shaping of the task (TASK);
- 2) scrolling of the results of task processing (LIST);
- 3) determination of the status of the task in the job queue (TEST).

One function can be performed in one run of the remote batch processing program.

After the remote processing program is called as a component of the display system of the Institute of Space Research, USSR Academy of Sciences, the following message is sent to the display console:

ENTER NAME OF BUILDER'S REGIME: LIST, TEST, TASK

Depending on the function selected, the user enters one of three possible replies: LIST, TEST, TASK. If the user does not require any of the functions offered component of the display system [sic], he enters the response [sic]

## 2. Formation of Tasks

Tasks are formed after entry of the function name "TASK." The user is accorded the opportunity to form a task and place it in the job queue. There is a set of regimes of the TASK function for making changes in a sequence of lines that has been entered. The regimes of the TASK function are defined by special statements. A distinguishing feature of the statements for this function is the presence of the symbols "%%." All lines entered, with the exception of the regimes of the TASK function, are given numbers. If the sequence of lines is altered (if lines are eliminated, replaced, or added), the line numbers are corrected. The numbers are set automatically.

As standard procedure, each line is assigned a number one larger than that of the last line.

## 3. Regimes of the TASK Function

### 1) %%ADDi

The regime %%ADD [sic] shows that the next line entered must be inserted in the existing sequence after the line numbered i.

2) %%REPi shows that the next line entered will replace the line numbered [i] in the existing sequence.

/6

### 3) %%DELi

The regime %%DEL indicates that the line numbered i must be eliminated from the existing sequence.



- |           |              |
|-----------|--------------|
| 4) %%LSTi | (format "a") |
| %%LSTj, l | (format "b") |
| %%LST     | (format "c") |

The regime %%LST is used to output to the display one or several lines of the existing sequence.

Depending on the regime statement format, either line i (format "a"), lines j through l (format "b"), or all lines (format "c") are output to the display.

- 5) %%END

The regime %%END interrupts execution of the TASK function. The task formulated is not placed in the job queue.

- 6) %%RUN

The %%RUN regime completes execution of the TASK function. The task formulated by the user is placed in the job queue. After this regime is implemented, the user is given the message:

YOUR TASK IS QUEUED. REMEMBER ITS NUMBER. IT IS iii  
[where] iii is the number of the task in the queue.

If the task cannot be placed in the queue (the queue has overflowed), the following message is sent:

YOUR TASK CANNOT BE QUEUED

#### 4. Viewing Results of Task Processing

After the user's task is processed in the subordinate section, the results of the job, output for system printing, are buffered in the direct access volume. The LIST function enables the user to view all buffered text on the display screen.

The entire text is divided into pages 7 alphanumeric printer lines long. Each line output for system printing is presented on the display screen either as two lines or as one line. The presentation format is determined by the user.

The user can view the text beginning at any position. The initial position from which text is viewed, and the presentation format are determined by LIST function regimes. /7

When the LIST function is called, the following message appears on the display screen: ENTER TASK NUMBER. The user can enter one of two replies:

```
n          n = 1 to 255  -- the number of the job whose results the
                        user wants to view.
```

```
END                -- complete job.
```

If the number of a job that has not yet been executed is entered, the following message appears in the first line on the screen:

BOOK IS NOT FOUND

After the number of an executed task is entered, the following message appears in the first line:

TASK nnn PAGE 000

Viewing of the listing sections required by the user is accomplished by entering control statements of the LIST function. When this is done, the first line on the screen always contains the following message:

contr. statement reaction TASK nnn PAGE jjj

where

contr. operator -- the name of the last control statement entered;  
 reaction -- the reaction to the last control statement entered:  
     a) spaces indicate that the statement has been executed;  
     b) INVALID REGIME--indicates that the control statement is not recognized or requires forbidden operations (e.g., scrolling through a page with a nonexistent number);  
     c) END OF TEXT FOUND--the end of the listing being viewed has been reached.  
 -- the number of the task whose listing is being viewed;  
 -- the number of the page to which the second line output to the screen belongs (as stated above, the first line always contains control information).

## 5. Control Statements of the LIST Function

/8

1) Fi      i = 0 to 57

The statement F changes the format for representing printout on the screen. If i = 0, one line of printout (121 bytes) is represented by two lines of display. The first line will contain 64 bytes, the second 57 bytes. It should be kept in mind that in the DOS/ES system the first byte is the printing control byte and does not appear in ordinary printout.

If in the control statement F i = 1 + 57, then 64 symbols of an alphanumeric printer line are presented in a single line of the screen, beginning with the i-th position. The statement F acts on all other control statements of the LIST function. If no F statement is entered, the system assumes F is 0.

2) Ji  $i = 0 \text{ to } 999$

The statement J causes 15 lines to be output to the screen, beginning with line 0 of the i-th page.

Note. One page contains 7 lines of printout.

3) Pi  $i = -999 \text{ to } 999$

The P statement causes 15 lines to be output to the screen, beginning with line number  $m + Fi$ , where  $m$  is the number of the top line when the P statement is entered. In other words, the P statement makes it possible to move the printout  $i$  pages forward (when  $i > 0$ ) or  $i$  pages back (when  $i < 0$ ) with respect to the current printout display on the screen.

Note. If  $i = 1$ , the statement may be entered in the form P.

4) Ti  $i = 0 \text{ to } 999$

The T regime makes it possible to output 15 lines to the screen, beginning with the i-th line.

5) Li  $i = -999 \text{ to } 999$

The L statement 15 lines to be output to the screen, beginning with line number  $i + m$ , where  $m$  is the number of the top line when the L statement is entered.

In other words, the L statement makes it possible to move the printout  $i$  lines forward ( $i > 0$ ) or back ( $i < 0$ ) with respect to the display present on the screen.

7) Z

/9

The Z statement makes it possible to output the last page of the printout to the screen.

8) END

The END statement makes it possible to complete execution of the LIST function.

9) PRINT i,j            i,j = 1,999

The PRINT statement makes it possible to output to an alphanumeric printer a portion of the printout, beginning with the i-th page and ending with the j-th. If an alphanumeric printer is inaccessible to the display system, the message NO PRINTER (printer unavailable) is output to the screen. If an alphanumeric printer is available for the display system, the screen is placed in the printer queue.

In any case, the PRINT statement terminates execution of the LIST function.

## 6. The TEST Function

The TEST function is used to obtain the status of a task.

The number of the task whose status must be determined is entered in response to the system prompt

ENTER TASK NUMBER

The number is entered in the form

n            n = 1 to 255

One of three possible replies is given:

YOUR TASK WAS NOT QUEUED

-- the task was not placed in the queue

YOUR TASK IS QUEUED AND  
NOT PROCESSED

-- the task is in the job queue,  
but has not been processed

YOUR TASK IS ALREADY PROCESSED

-- the task has been processed

## ALPHABETICAL LIST OF MESSAGES IN THE REMOTE BATCH PROCESSING SUBSYSTEM

/10

Text of message	Function performed	Cause	Programmer's actions
1	2	3	4
BOOK IS NOT FOUND	LIST	Result of task with requested number not found.	Check correctness of task number; if number is correct, await processing of task.
END OF TEXT FOUND	LIST	End of listing being viewed is found.	Enter a J or Z statement to view the required page; the P, L, and T statements cannot be entered.
ENTER NAME OF BUILDER'S REGIME: TASK, LIST, TEST		The activated BUILDER subsystem requests name of function.	Enter name of required BUILDER subsystem function: LIST, TEST, TASK, END.
ENTER TASK NUMBER	LIST	LIST requests task number.	Enter number of task whose results are to be viewed.
ENTER TASK NUMBER	TEST	TEST function requests number of task whose status must be determined.	Enter number of task whose status must be determined.
ERROR CODE 1		Unrecognized function selection statement.	Enter correct statement: TASK, LIST, TEST, END.
ERROR CODE 1	TASK	Program text too long.	Catalog part of program into source module library so that the statement %%COPY can be used later.
ERROR CODE 2	TASK	Unrecognized regime of TASK function.	Enter correct regime of TASK function.
ERROR CODE 3	TASK	Incorrect syntax in control statement of TASK function.	Enter syntactically correct statement.

/11

1	2	3	4	/12
ERROR CODE 4	TASK	Overflow of system work file.	Wait until work file is freed of tasks from other screens, or shorten text of program.	
INVALID REGIME	LIST	Unrecognized or syntactically incorrect control statement.	Enter correct control statement.	
YOUR TASK CANNOT BE QUEUED*	TASK	Task cannot be queued because of queue overflow.	Not required.	
YOUR TASK IS ALREADY PROCESSED*	TEST	Task already processed.	Not required.	
YOUR TASK IS QUEUED AND NOT PROCESSED*	TEST	Task is in job queue but not yet processed.	Not required.	
YOUR TASK IS QUEUED. REMEMBER ITS NUMBER. IT IS NNN*	TASK	Task placed in job queue. Its number is NNN.	Remember task number for subsequent inquiries about it and for viewing results.	/13
YOUR TASK WAS NOT QUEUED*	TEST	Task not placed in queue.	Reactivate BUILDER subsystem and enter correct task number.	

#### Note

Messages marked with an asterisk ("\*") terminate operation of the BUILDER subsystem and stay on the screen for 5 seconds.



## Supplement

/14

### Program for Printing Results of Task Processing under Control of Remote Batch Processing Subsystem

The program for printing the results of task processing under the control of the remote batch processing subsystem is a program independent of the display system. Under this program results are printed in batch mode under the control of the DOS ES system. The name of the calling phase of the print program is BUILDPRRT.

The print-results program has one control statement: PRINT.

The statement format is PRINT I [where] I is the number of the task whose results are to be output for printing.

Any number of PRINT statements can be processed in a single run of the print program. If an invalid statement appears in the print-results program or if the results of the processing of the task having the required number are not available, an error message is output for system printing, and the program run continues. The program stops running when there is an "End of File" status at the job input device. The results of the program run are culled from the file bearing the program name LISYSSL 1.

Sample task for running the print-results program.

```
//JOB 222 PRINT CHESALIN
//ASSGN SYSSLB.X'192' DISK 203
//PAUSE
//DLBL LISYSSL.'SLB2'
//EXEC BUILDPRRT
  PRINT 1
  PRINT 023
/*
/*
```